

Name

intro – introduction to Yellow Pages (YP) library functions

Description

This section describes those functions that are in the Yellow Pages library.

getnetgrent (3yp)

Name

getnetgrent, setnetgrent, endnetgrent, innetgr – get network group entry

Syntax

```
innetgr(netgroup, machine, user, domain)
char *netgroup, *machine, *user, *domain;
```

```
setnetgrent(netgroup)
char *netgroup
```

```
endnetgrent()
```

```
getnetgrent(machinep, userp, domainp)
char **machinep, **userp, **domainp;
```

Description

The `innetgr` routine accesses the `netgroup` file and checks to see if the specified input parameters match an entry in the file. The routine returns 1 if it matches an entry, or 0 if it does not. Any of the three strings; **machine**, **user**, or **domain** can be NULL, which signifies any string in that position is valid.

The `getnetgrent` routine returns the next member of a network group. After the call, `machinep` will contain a pointer to a string containing the name of the machine part of the network group member, and similarly for `userp` and `domainp`. If **machinep**, **userp** or **domainp** is returned as a NULL pointer, it signifies any string is valid. The `getnetgrent` routine allocates space for the name by using the `malloc` routine. This space is released when an `endnetgrent` call is made. The `getnetgrent` routine returns 1 if it succeeds in obtaining another member of the network group, or 0 if it reaches the end of the group.

The `setnetgrent` routine establishes the network group from which `getnetgrent` will obtain members, and also restarts calls to `getnetgrent` from the beginning of the list. If the previous `setnetgrent` call was to a different network group, an `endnetgrent` call is implied.

The `endnetgrent` routine releases the space allocated during the `getnetgrent` calls.

Files

```
/etc/netgroup
/etc/yp/domain/netgroup
/etc/yp/domain/netgroup.byuser
/etc/yp/domain/netgroup.byhost
```


Name

yp_get_default_domain, yp_bind, yp_unbind, yp_match, yp_first, yp_next, yp_all,
yp_order, yp_master, yperr_string, ypprot_err – Yellow Pages client package

Syntax

```
#include <rpcsvc/ypclnt.h>
```

```
yp_get_default_domain(outdomain)
char **outdomain;
```

```
yp_bind(indomain)
char *indomain;
```

```
void yp_unbind(indomain)
char *indomain;
```

```
yp_match(indomain, inmap, inkey, inkeylen, outval, outvallen)
char *indomain;
char *inmap;
char *inkey;
int inkeylen;
char **outval;
int *outvallen;
```

```
yp_first(indomain, inmap, outkey, outkeylen, outval, outvallen)
char *indomain;
char *inmap;
char **outkey;
int *outkeylen;
char **outval;
int *outvallen;
```

```
yp_next(indomain, inmap, inkey, inkeylen, outkey, outkeylen,
char *indomain;
char *inmap;
char *inkey;
int inkeylen;
char **outkey;
int *outkeylen;
char **outval;
int *outvallen;
```

```
yp_all(indomain, inmap, incallback)
char *indomain;
char *inmap;
struct ypall_callback incallback;
```

```
yp_order(indomain, inmap, outorder)
char *indomain;
char *inmap;
int *outorder;
```

```
yp_master(indomain, inmap, outname)
char *indomain;
char *inmap;
```

ypclnt(3yp)

```
char **outname;  
char *yperr_string(incode)  
int incode;  
ypprot_err(incode)  
unsigned int incode;
```

Description

This package of functions provides an interface to the Yellow Pages (YP) data base lookup service. The package can be loaded from the standard library, `/lib/libc.a`. Refer to `ypfiles(5yp)` and `ypserv(8yp)` for an overview of the Yellow Pages, including the definitions of **map** and **domain**, and for a description of the servers, data bases, and commands that constitute the YP application.

All input parameters names begin with **in**. Output parameters begin with **out**. Output parameters of type **char **** should be addresses of uninitialized character pointers. The YP client package allocates memory using `malloc(3)`. This memory can be freed if the user code has no continuing need for it. For each **outkey** and **outval**, two extra bytes of memory are allocated at the end that contain NEWLINE and NULL, respectively, but these two bytes are not reflected in **outkeylen** or **outvallen**. The **indomain** and **inmap** strings must be non-null and null-terminated. String parameters that are accompanied by a count parameter cannot be null, but can point to null strings, with the count parameter indicating this. Counted strings need not be null-terminated.

All functions of type **int** return 0 if they succeed, or a failure code (`YPERR_ xxxx`) if they do not succeed. Failure codes are described under **Diagnostics**.

The YP lookup calls require a map name and a domain name. It is assumed that the client process knows the name of the map of interest. Client processes fetch the node's default domain by calling `yp_get_default_domain`, and use the returned **outdomain** as the **indomain** parameter to successive YP calls.

To use YP services, the client process must be bound to a YP server that serves the appropriate domain. The binding is accomplished with `yp_bind`. Binding need not be done explicitly by user code; it is done automatically whenever a YP lookup function is called. The `yp_bind` function can be called directly for processes that make use of a backup strategy in cases when YP services are not available.

Each binding allocates one client process socket descriptor; each bound domain requires one socket descriptor. Multiple requests to the same domain use that same descriptor. The `yp_unbind` function is available at the client interface for processes that explicitly manage their socket descriptors while accessing multiple domains. The call to `yp_unbind` makes the domain unbound, and frees all per-process and per-node resources used to bind it.

If an RPC failure results upon use of a binding, that domain will be unbound automatically. At that point, the `ypclnt` layer will retry forever or until the operation succeeds. This action occurs provided that `ypbind` is running, and either the client process cannot bind a server for the proper domain, or RPC requests to the server fail.

ypclnt(3yp)

The `ypbind -s` option allows the system administrator to lock `ypbind` to a particular domain and set of servers. Up to four servers can be specified. An example of the `-s` option follows:

```
/etc/ypbind -s domain,server1[,server2,server3,server4]
```

The `ypclnt` layer will return control to the user code, either with an error code, or with a success code and any results under certain circumstances. For example, control will be returned to the user code when an error is not RPC-related and also when the `ypbind` function is not running. An additional situation that will cause the return of control is when a bound `ypserv` process returns any answer (success or failure).

The `yp_match` function returns the value associated with a passed key. This key must be exact; no pattern matching is available.

The `yp_first` function returns the first key-value pair from the named map in the named domain.

The `yp_next` function returns the next key-value pair in a named map. The `inkey` parameter should be the **outkey** returned from an initial call to `yp_first` (to get the second key-value pair) or the one returned from the `n`th call to `yp_next` (to get the `n`th + second key-value pair).

The concept of first and of next is particular to the structure of the YP map being processed; there is no relation in retrieval order to either the lexical order within any original (non-YP) data base, or to any obvious numerical sorting order on the keys, values, or key-value pairs. The only ordering guarantee made is that if the `yp_first` function is called on a particular map, and then the `yp_next` function is repeatedly called on the same map at the same server until the call fails with a reason of `YPERR_NOMORE`, every entry in the data base will be seen exactly once. Further, if the same sequence of operations is performed on the same map at the same server, the entries will be seen in the same order.

Under conditions of heavy server load or server failure, it is possible for the domain to become unbound, then bound once again (perhaps to a different server) while a client is running. This can cause a break in one of the enumeration rules; specific entries may be seen twice by the client, or not at all. This approach protects the client from error messages that would otherwise be returned in the midst of the enumeration. Enumerating all entries in a map is accomplished with the `yp_all` function.

The `yp_all` function provides a way to transfer an entire map from server to client in a single request using TCP (rather than UDP as with other functions in this package). The entire transaction takes place as a single RPC request and response. The `yp_all` function can be used like any other YP procedure, to identify the map in the normal manner, and to supply the name of a function that will be called to process each key-value pair within the map. Returns from the call to `yp_all` occur only when the transaction is completed (successfully or unsuccessfully), or when the `foreach` function decides that it does not want to see any more key-value pairs.

The third parameter to `yp_all` is

```
struct ypall_callback *incallback {  
    int (*foreach) ();  
    char *data;  
};
```


ypclnt(3yp)

The function `foreach` is called

```
foreach(instatus, inkey, inkeylen, inval, invallen, indata);
int instatus;
char *inkey;
int inkeylen;
char *inval;
int invallen;
char *indata;
```

The **instatus** parameter will hold one of the return status values defined in `<rpcsvc/yp_prot.h>` — either `YP_TRUE` or an error code. (See **ypprot_err**, below, for a function that converts a YP protocol error code to a `ypclnt` layer error code.)

The key and value parameters are somewhat different than defined in the syntax section above. First, the memory pointed to by the **inkey** and **inval** parameters is private to the `yp_all` function and is overwritten with the arrival of each new key-value pair. It is the responsibility of the `foreach` function to do something useful with the contents of that memory, but it does not own the memory itself. Key and value objects presented to the `foreach` function look exactly as they do in the server's map — if they were not newline-terminated or null-terminated in the map, they will not be here either.

The **indata** parameter is the contents of the **incallback->data** element passed to `yp_all`. The **data** element of the callback structure may be used to share state information between the `foreach` function and the mainline code. Its use is optional, and no part of the YP client package inspects its contents.

The `foreach` function returns a Boolean value. It should return zero to indicate that it wants to be called again for further received key-value pairs, or nonzero to stop the flow of key-value pairs. If `foreach` returns a nonzero value, it is not called again; the functional value of `yp_all` is then 0.

The `yp_order` function returns the order number for a map.

The `yp_master` function returns the machine name of the master YP server for a map.

The `yperr_string` function returns a pointer to an error message string that is null-terminated but contains no period or new line.

The `ypprot_err` function takes a YP protocol error code as input and returns a `ypclnt` layer error code, which may be used in turn as an input to `yperr_string`.

Diagnostics

All integer functions return 0 if the requested operation is successful, or one of the following errors if the operation fails.

<code>#define YPERR_BADARGS</code>	1	<code>/* args to function are bad */</code>
<code>#define YPERR_RPC</code>	2	<code>/* RPC failure - domain has been unbound */</code>
<code>#define YPERR_DOMAIN</code>	3	<code>/* can't bind to server on this domain */</code>
<code>#define YPERR_MAP</code>	4	<code>/* no such map in server's domain */</code>
<code>#define YPERR_KEY</code>	5	<code>/* no such key in map */</code>
<code>#define YPERR_YPERR</code>	6	<code>/* internal yp server or client error */</code>
<code>#define YPERR_RESRC</code>	7	<code>/* resource allocation failure */</code>
<code>#define YPERR_NOMORE</code>	8	<code>/* no more records in map database */</code>
<code>#define YPERR_PMAP</code>	9	<code>/* can't communicate with portmapper */</code>

ypclnt(3yp)

```
#define YPERR_YPBIND    10 /* can't communicate with ypbind */
#define YPERR_YPSESV    11 /* can't communicate with ypserv */
#define YPERR_NODOM     12 /* local domain name not set */
```

Files

/usr/include/rpcsvc/ypclnt.h
/usr/include/rpcsvc/yp_prot.h

See Also

ypfiles(5yp), ypserv(8yp)

yppasswd(3yp)

Name

yppasswd – update user password in yellow pages password map.

Syntax

```
#include <rpcsvc/yppasswd.h>
```

```
yppasswd(oldpass, newpw)
```

```
char *oldpass;
```

```
struct passwd *newpw;
```

Description

The `yppasswd` routine uses Remote Procedure Call (RPC) and External Data Representation (XDR) routines to update a user password in a Yellow Pages password map. The RPC and XDR elements that are used are listed below under the RPC INFO heading.

If *oldpass* is indeed the old user password, this routine replaces the password entry with *newpw*. It returns 0 if successful.

RPC Information

program number:

YPPASSWDPROG

xdr routines:

xdr_yppasswd(xdrs, yp)

XDR *xdrs;

struct yppasswd *yp;

xdr_yppasswd(xdrs, pw)

XDR *xdrs;

struct passwd *pw;

procs:

YPPASSWDPROC_UPDATE

Takes *struct yppasswd* as argument, returns integer.

Same behavior as *yppasswd()* wrapper.

Uses UNIX authentication.

versions:

YPPASSWDVERS_ORIG

structures:

```
struct yppasswd {
```

```
    char *oldpass; /* old (unencrypted) password */
```

```
    struct passwd newpw; /* new pw structure */
```

```
};
```

See Also

yppasswd(1yp), yppasswdd(8yp)

A

- abort subroutine (standard C), 3-6
- abs subroutine (standard C), 3-7
- absolute value function, 3-7, 3-343, 3-345
- acos subroutine, 3-363
- acos subroutine (math), 3-364
- acosh subroutine, 3-335
- addch macro, 3-200
 - See also* putchar subroutine
- addch subroutine, 3-528
- addstr macro, 3-202
- addstr subroutine, 3-528
- alarm subroutine (standard C), 3-8
 - See also* sleep subroutine (standard C)
- alloca subroutine, 3-84, 3-86
- arc subroutine, 3-559
- arccosine function, 3-364
- arcsine function, 3-364
- arctangent function, 3-364
- argument list
 - portable procedures for variable, 3-186
- argument vector
 - getting option letter, 3-61
- ASCII character
 - classifying, 3-25
- ASCII string
 - converting long integer to, 3-5
 - converting to, 3-34
 - converting to numbers, 3-10
- asctime subroutine (standard C), 3-20
- asin subroutine, 3-362
- asin subroutine (math), 3-364
- asinh subroutine, 3-335

- assert macro, 3-9
- atan subroutine, 3-363
- atan subroutine (math), 3-364
- atan2 subroutine, 3-363
- atan2 subroutine (math), 3-364
- atanh subroutine, 3-335
- atof subroutine (standard C), 3-10
 - isalpha subroutine, 3-10
- atoi subroutine, 3-10
- atol subroutine, 3-10
- attroff macro, 3-203
- attron macro, 3-203
- attrset macro, 3-203
- auth database (general)
 - getting/setting entry, 3-51
- a64l subroutine, 3-5

B

- baudrate subroutine, 3-205
- bcmp subroutine, 3-14
- bcopy subroutine, 3-14
- beep subroutine, 3-206
- bessel keyword (math), 3-336
- binary search routine, 3-13
 - managing tree searches, 3-179
- bit
 - determining setting in byte, 3-14
- box subroutine, 3-207, 3-528
- bsearch subroutine, 3-13
 - tsearch subroutine, 3-13
- bstring keyword, 3-14
- buffering
 - types of, 3-517

byte

swapping with PDP-11s, 3-171

byte sex, 3-58

byteorder keyword, 3-372

bzero subroutine, 3-14

C

C library

See also libc library

cabs function, 3-348

cabs subroutine, 3-349

calloc subroutine (special library), 3-555

calloc subroutine (standard C), 3-84, 3-86

catclose subroutine, 3-281

catgetmsg subroutine, 3-279

catgets subroutine, 3-280

catopen subroutine, 3-281

cbreak subroutine, 3-208

ceil subroutine, 3-343, 3-345

ceiling function

returning integer, 3-343, 3-345

circle subroutine, 3-559

clear macro, 3-209

clear subroutine, 3-528

clearerr subroutine, 3-498

clearok subroutine, 3-210, 3-528

clock subroutine, 3-15

closedir subroutine, 3-27

closelog subroutine, 3-175

closepl subroutine, 3-559

clrtoobot macro, 3-211

clrtoobot subroutine, 3-528

clrtoeol macro, 3-212

clrtoeol subroutine, 3-528

connect request

determining status, 3-602

listening for, 3-590

connection

receiving expedited data, 3-600

receiving normal data, 3-600

connectionless mode

receiving data, 3-609

receiving error information, 3-611

connectionless mode (cont.)

sending data, 3-620

cont subroutine, 3-559

cos subroutine, 3-362

cos subroutine (math), 3-364

cosh subroutine (math), 3-367

cosine function, 3-364

creatediskbyname subroutine, 3-526

crmode subroutine, 3-528

crypt subroutine

encryption, 3-18

ctermid subroutine, 3-495

See also ttyname subroutine

compared with ttyname subroutine, 3-495

ctime subroutine (standard C), 3-20

System V and, 3-23

ctype keyword, 3-25

curses library, 3-193

curses package, 3-193, 3-528

main routines, 3-193

subroutine list, 3-528

cursor

optimizing movement, 3-193, 3-528

cuserid subroutine, 3-496

See also getlogin subroutine

D

data base subroutine, 3-530

Data Encryption Standard, 3-18

date

converting to ASCII, 3-20

getting, 3-177

dbm keyword, 3-530

dbminit subroutine, 3-530

restrictions, 3-531

def_prog_mode subroutine, 3-213

See also reset_prog_mode subroutine

def_shell_mode subroutine, 3-213

See also reset_shell_mode subroutine

delay_output subroutine, 3-214

delch subroutine, 3-215, 3-528

delete subroutine, 3-530

deleteln subroutine, 3-216, 3-528

delwin subroutine, 3-217, 3-528

directory

See also working directory

descending tree, 3-49

operations, 3-27

scanning, 3-120

directory keyword, 3-27

disconnect

retrieving information, 3-605

disk

getting description, 3-526, 3-533

div subroutine (ANSI C), 3-30

doupdate subroutine, 3-275

draino subroutine, 3-218

drand48 subroutine, 3-31

E

echo subroutine, 3-219, 3-528

ecvt subroutine, 3-34

edata subroutine, 3-37, 3-38

effective group ID

setting, 3-129

effective user ID

setting, 3-129

encryption

crypt subroutine, 3-18

end subroutine, 3-37, 3-38

endauthent subroutine, 3-51

endfsent subroutine, 3-534

endgrent subroutine, 3-56

endhostent subroutine, 3-373

endnetent subroutine, 3-375

endnetgrent subroutine, 3-626

endprotoent subroutine, 3-377

endpwent subroutine, 3-65

endrpcent subroutine, 3-67

endservent subroutine, 3-379

endttyent subroutine, 3-71

endwin subroutine, 3-220, 3-528

environ subroutine, 3-39, 3-41

environment

changing, 3-109

environment (cont.)

getting variable values, 3-55

environment (POSIX)

See POSIX environment

environment (System V)

See System V environment

erand48 subroutine, 3-31

erase macro (curses), 3-221

erase subroutine (curses), 3-528

erase subroutine (plot), 3-559

erasechar subroutine, 3-222

erf function, 3-337

erf subroutine, 3-338

erfc function, 3-337

erfc subroutine, 3-338

error function, 3-338

System V and, 3-360

error message (system)

getting, 3-105

error messages

transport function, 3-581

error_c_get_text, 3-399

error_c_text, 3-400

etext subroutine, 3-37, 3-38

Euclidean distance, 3-349

execl subroutine, 3-39, 3-41

execle subroutine, 3-39, 3-41

execvp subroutine, 3-39, 3-41

execv subroutine, 3-39, 3-41

execv subroutine, 3-39, 3-41

execve system call

See also execl subroutine

execvp subroutine, 3-39, 3-41

diagnostics, 3-40, 3-42

restricted, 3-40

restrictions, 3-42

_exit subroutine, 3-43

exit subroutine (standard C), 3-43

exp function, 3-339

exp subroutine (math), 3-341

erf subroutine, 3-341

System V and, 3-342

expm1 function, 3-339

expm1 subroutine (math), 3-341

exponent

splitting into, 3-46

exponential function, 3-341

F

fabs subroutine, 3-343, 3-345

Fault Management

pfm_cleanup, 3-420

pfm_enable, 3-422

pfm_enable_faults, 3-423

pfm_inhibit, 3-424

pfm_inhibit_faults, 3-425

pfm_init, 3-426

pfm_reset_cleanup, 3-427

pfm_rls_cleanup, 3-428

pfm_signal, 3-430

fclose subroutine, 3-497

fcntl system call

See also lockf subroutine

fcvt subroutine, 3-34

fdopen subroutine, 3-500

feof subroutine, 3-498

ferror subroutine, 3-498

fetch subroutine, 3-530

fexp function, 3-339

fexpm1 function, 3-339

fflush subroutine, 3-497

ffs subroutine, 3-14

fgetc subroutine (standard I/O), 3-504

fgets subroutine, 3-505

file

See also temporary file

executing, 3-39, 3-41

locking region, 3-80

setting access time, 3-183

setting modification time, 3-183

file name

making unique, 3-92

file system

getting information on mounted, 3-60

maintaining, 3-79

fileno subroutine, 3-498

files

remove, 3-116

firstkey subroutine, 3-530

flash subroutine, 3-206

floating point remainder, 3-343, 3-345

flog function, 3-339

flog1p function, 3-339

flog10 function, 3-339

floor function

returning integer, 3-343, 3-345

floor subroutine, 3-345

flushinp subroutine, 3-223

fopen subroutine, 3-500

System V and, 3-501

formatted input

converting, 3-512

formatted output

printing, 3-506

printing from argument list, 3-523

fpathconf subroutine, 3-102

fp_class routine

RISC only, 3-45

fprintf function, 3-506

fprintf subroutine, 3-288

fputc subroutine, 3-510

fputs subroutine, 3-511

fread subroutine, 3-502

free subroutine (special library), 3-555

free subroutine (standard C), 3-84, 3-86

freopen subroutine, 3-500

frexp subroutine, 3-46

fscanf function, 3-512

fscanf subroutine, 3-290

fseek subroutine (standard I/O), 3-503

fstab file

See also getfsent subroutine

closing, 3-534

getting entry, 3-534

getting file system name, 3-534

getting file system type, 3-534

getting special file name, 3-534

setting, 3-534

ftell subroutine (standard I/O), 3-503
 ftime subroutine, 3-177
 ftok subroutine, 3-48
 ftw subroutine, 3-49
 fwrite subroutine, 3-502

G

gamma function, 3-346, 3-347
 gamma subroutine, 3-346, 3-347
 gcvf subroutine, 3-34
 getauthuid subroutine, 3-51
 getc subroutine (standard I/O), 3-504
 See also ungetc subroutine
 getcap subroutine, 3-528
 getch macro, 3-224
 getch subroutine, 3-528
 See also nodelay subroutine
 getchar subroutine, 3-504
 getcwd subroutine (standard C), 3-53
 getdiskbyname subroutine, 3-533
 getenv subroutine (standard C), 3-55
 See also putenv subroutine
 getfsent subroutine, 3-534
 getfsfile subroutine, 3-534
 getfsspec subroutine, 3-534
 getfstype subroutine, 3-534
 getgrent subroutine, 3-56
 restrictions, 3-57
 getgrgid subroutine, 3-56
 getgrnam subroutine, 3-56
 gethostbyaddr subroutine, 3-373
 gethostbyname subroutine, 3-373
 gethostent subroutine, 3-373, 3-373
 restrictions, 3-374
 svc.conf file and, 3-373
 gethostsex(3) reference page, 3-58
 getlogin subroutine, 3-59
 getmountent subroutine, 3-60
 See also statfs subroutine
 getnetbyaddr subroutine, 3-375
 getnetbyname subroutine, 3-375
 getnetent subroutine, 3-375
 restrictions, 3-376

getnetgrent subroutine, 3-626
 getopt subroutine, 3-61
 getpass subroutine, 3-63
 See also crypt subroutine
 getprotobyname subroutine, 3-377
 getprotoent subroutine, 3-377
 restrictions, 3-378
 getpw subroutine, 3-64
 getpwuid subroutine, 3-64
 getpwent subroutine, 3-65
 See also putpwent subroutine
 restrictions, 3-66
 getpwnam subroutine, 3-65
 getpwuid subroutine, 3-65
 getrpcbyname subroutine, 3-67
 getrpcbynumber subroutine, 3-67
 getrpcent subroutine, 3-67
 restrictions, 3-67
 gets subroutine, 3-505
 getservbyname subroutine, 3-379
 getservbyport subroutine, 3-379
 getservent subroutine, 3-379
 restrictions, 3-380
 getstr macro, 3-226
 getstr subroutine, 3-528
 getsvc call, 3-69
 gettmode subroutine, 3-528
 getttyent subroutine, 3-71
 restrictions, 3-72
 getttynam subroutine, 3-71
 getw subroutine, 3-504
 getwd subroutine, 3-73
 getyx macro, 3-228
 getyx subroutine, 3-528
 gmtime subroutine (standard C), 3-20
 graphics interface, 3-559
 group access list
 initializing, 3-536
 group file (general)
 getting entry, 3-56, 3-67
 gtty subroutine, 3-170

H

hash table search routine, 3-76

has_ic subroutine, 3-229

See also delch subroutine

See also insch subroutine

has_il subroutine, 3-230

See also deleteln subroutine

See also insertln macro

hcreate subroutine, 3-76

hdestroy subroutine, 3-76

hes_error routine, 3-74

hes_init routine, 3-74

hesiod, 3-74

hes_resolve routine, 3-74

hes_to_bind routine, 3-74

hosts file

getting entry, 3-373

hsearch subroutine, 3-76

restrictions, 3-76

htonl subroutine, 3-372

htons subroutine, 3-372

hyperbolic function, 3-367

inverse, 3-335

hypot function, 3-348

hypot subroutine, 3-349

I

idlok subroutine, 3-231

inch macro, 3-232

inch subroutine, 3-528

index subroutine (standard C), 3-147

inet keyword, 3-381

inet_addr subroutine, 3-381

inet_lnaof subroutine, 3-381

inet_makeaddr subroutine, 3-381

inet_netof subroutine, 3-381

inet_network subroutine, 3-381

inet_ntoa subroutine, 3-381

initgroups subroutine, 3-536

initscr subroutine, 3-233, 3-528

See also newterm subroutine

See also refresh macro

initstate subroutine, 3-113

innetgr subroutine, 3-626

insch macro, 3-234

insch subroutine, 3-528

insertln macro, 3-235

insertln subroutine, 3-528

insque subroutine, 3-77

Interface to the Location Broker

lb_register, 3-417

lb_unregister, 3-418

Interface to the Remote Procedure Call, 3-434

rpc_alloc_handle, 3-432

rpc_bind, 3-436

rpc_clear_binding, 3-438

rpc_clear_server_binding, 3-440

rpc_free_handle, 3-443

rpc_inq_binding, 3-444

rpc_inq_object, 3-446

rpc_name_to_sockaddr, 3-451

rpc_register, 3-452

rpc_register_mgr, 3-454

rpc_register_object, 3-456

rpc_set_async_ack, 3-458

rpc_set_binding, 3-460

rpc_set_fault_mode, 3-462

rpc_set_short_timeout, 3-463

rpc_shutdown, 3-464

rpc_sockaddr_to_name, 3-465

rpc_unregister, 3-467

rpc_use_family, 3-469

rpc_use_famfyl_wk, 3-471

international subroutines

introduction, 3-277

Internet address

manipulation routines, 3-381

specifying, 3-381

interprocess communication facility

See IPC

interprocess communication package, 3-48

intrflush subroutine, 3-236

intro(3) keyword, 3-1

intro(3cur) keyword, 3-193

intro(3m) keyword, 3-333

intro(3n) keyword, 3-369

intro(3x) keyword, 3-525

intro(3yp) keyword, 3-625

IPC

library functions, 3-369

isalnum subroutine, 3-25

isalpha subroutine, 3-25

atof subroutine (standard C), 3-10

isascii subroutine, 3-25

isatty subroutine (standard C), 3-181

isctrl subroutine, 3-25

isdigit subroutine, 3-25

isgraph subroutine, 3-25

islower subroutine, 3-25

isprint subroutine, 3-25

ispunct subroutine, 3-25

isspace subroutine, 3-25

isupper subroutine, 3-25

J

j0 subroutine, 3-336

j1 subroutine, 3-336

jn subroutine, 3-336

jrand48 subroutine, 3-31

K

Kerberos files

intro.3krb, 3-295

Kerberos routines

acl_add, 3-299

acl_canonicalize_principal, 3-299

acl_check, 3-299

acl_delete, 3-299

acl_exact_match, 3-299

acl_initialize, 3-299

des_crypt, 3-303

des_key_sched, 3-303

des_quad_cksum, 3-303

des_string_to_key, 3-303

kerberos 3, 3-307

kname_parse, 3-299

krb_get_cred, 3-307

Kerberos routines (cont.)

krb_get_lrealm, 3-312

krb_get_phost, 3-312

krb_get_pw_in_tkt, 3-328

krb_get_svc_in_tkt, 3-328

krb_mk_req, 3-307

krb_mk_safe, 3-307

krb_rd_req, 3-307

krb_rd_safe, 3-307

krb_recvauth, 3-316

krb_recvmutual, 3-323

krb_sendauth, 3-316

krb_sendmutual, 3-323

krb_set_srvtab_string, 3-326

krb_set_tkt_string, 3-326

krb_svc_init, 3-328

keypad subroutine, 3-237

kill system call

See also pause subroutine

killchar subroutine, 3-238

L

l3tol subroutine, 3-79

label subroutine, 3-559

labs subroutine (ANSI C), 3-7

lb_lookup_object, 3-401, 3-405

lb_lookup_object_local, 3-408

lb_lookup_range, 3-411

lb_lookup_type, 3-414

lcong48 subroutine, 3-31

ldexp subroutine, 3-46

ldiv subroutine (ANSI C), 3-30

leaveok subroutine, 3-239, 3-528

lfind subroutine, 3-82

libc library

contents, 3-1

libm library

contents, 3-2

diagnostics and, 3-4

introduction, 3-333

library

functions, 3-1

- library function**
 - miscellaneous, 3-525
- line subroutine (plot),** 3-559
- linear search routine,** 3-82
- linemod subroutine,** 3-559
- localtime subroutine (standard C),** 3-20
- lockf subroutine,** 3-80
 - diagnostics, 3-81
 - restrictions, 3-81
- log function,** 3-339
- log subroutine (math),** 3-341
- log1p function,** 3-339
- log1p subroutine (math),** 3-341
- log10 function,** 3-339
- log10 subroutine (math),** 3-341
- logarithm function,** 3-341
- login name**
 - getting, 3-59
 - getting character-string representation, 3-496
- longjmp subroutine,** 3-123
- longname subroutine,** 3-240, 3-528
- lrnd48 subroutine,** 3-31
- lsearch subroutine,** 3-82
- ltol3 subroutine,** 3-79
- l64a subroutine,** 3-5

M

- mallinfo subroutine,** 3-555
 - structure returned, 3-556e
- malloc subroutine (special library),** 3-555 to 3-557
 - restrictions, 3-556
- malloc subroutine (standard C),** 3-84, 3-86
 - See also* malloc subroutine (special library)
 - restricted, 3-85
 - restrictions, 3-87
- mallop subroutine,** 3-555
- mantissa**
 - splitting into, 3-46
- math library**
 - See* libm library
- matherr subroutine,** 3-360
 - error-handling procedures, 3-361t
 - using, 3-360e

- memccpy subroutine,** 3-88
 - restrictions, 3-89
- memchr subroutine,** 3-88
- memcmp subroutine,** 3-88
- memcpy subroutine,** 3-88
- memmove subroutine,** 3-88
- memory**
 - See also* shared memory
 - allocating, 3-84, 3-86, 3-185, 3-555 to 3-557, 3-572
 - freeing, 3-583
 - operations, 3-88
 - shared memory, 3-86
- memory area**
 - defined, 3-88
- memory keyword,** 3-88
- memset subroutine,** 3-88
- meta subroutine,** 3-241
- mkfifo function,** 3-90
- mktemp subroutine,** 3-92
- modf subroutine,** 3-46
- moncontrol subroutine,** 3-95
- monitor subroutine,** 3-95
- monstartup subroutine,** 3-95
- move macro (curses),** 3-242
- move subroutine (curses),** 3-528
- move subroutine (plot),** 3-559
- mrnd48 subroutine,** 3-31
- mvaddch macro,** 3-200
- mvaddstr macro,** 3-202
- mvcur subroutine,** 3-243, 3-528
- mvdelch subroutine,** 3-215
- mvgetch macro,** 3-224
- mvgetstr macro,** 3-226
- mvinch macro,** 3-232
- mvinsch macro,** 3-234
- mvprintw subroutine,** 3-253
- mvscanw subroutine,** 3-260
- mvwaddch macro,** 3-200
- mvwaddstr macro,** 3-202
- mvwdelch subroutine,** 3-215
- mvwgetch macro,** 3-224
- mvwgetstr macro,** 3-226

mvwin subroutine, 3-244
mvwinch macro, 3-232
mvwisch macro, 3-234
mvwprintw subroutine, 3-253
mvwscanw subroutine, 3-260

N

name list
 defined, 3-100, 3-101
napms subroutine, 3-245
NBS Data Encryption Standard, 3-18
NCS commands and library routines, 3-387
netgroup file
 See also inetgr subroutine
 getting member entry, 3-626
networks file
 getting entry, 3-375
newpad subroutine, 3-246
newterm subroutine, 3-247
newwin subroutine, 3-248, 3-528
nextkey subroutine, 3-530
nice subroutine, 3-99
nl macro, 3-249
nl subroutine, 3-528
nl_fprintf subroutine, 3-284
nl_fscanf subroutine, 3-286
nlist subroutine, 3-100, 3-101
nl_langinfo subroutine, 3-282
nl_printf subroutine, 3-284
nl_scanf subroutine, 3-286
nl_sprintf subroutine, 3-284
nl_sscanf subroutine, 3-286
nocbreak subroutine, 3-208
nocrmode subroutine, 3-528
nodelay subroutine, 3-250
noecho subroutine, 3-219, 3-528
nonl subroutine, 3-249, 3-528
noraw subroutine, 3-255, 3-528
nrand48 subroutine, 3-31
ntohl subroutine, 3-372
ntohs subroutine, 3-372

O

opendir subroutine, 3-27, 3-28e
openlog subroutine, 3-175
openpl subroutine, 3-559
 See also plot file

Operations on Socket Addresses

socket_equal, 3-477
socket_family_from_name, 3-479
socket_family_to_name, 3-480
socket_from_name, 3-481
socket_to_name, 3-483
socket_to_numeric_name, 3-484
socket_valid_families, 3-486
socket_valid_family, 3-487

Operations on Universal Unique Identifiers

uuid_equal, 3-490

Operations on Universal Unique Identifiers

uid_encode, 3-489
uid_decode, 3-488
uuid_gen, 3-491

orderly release

acknowledging receipt, 3-607

overlay subroutine, 3-251, 3-528

overwrite subroutine, 3-251, 3-528

P

packetfilter

pfopen subroutine, 3-106

password

reading, 3-63
updating in YP map, 3-632

password file (general)

getting entry, 3-65
getting name from, 3-64
writing entry, 3-110

pathconf subroutine, 3-102

pause subroutine, 3-104

pclose subroutine, 3-107

perror subroutine (standard C), 3-105

pfopen subroutine, 3-106

plot keyword, 3-559

pnoutrefresh subroutine, 3-252

point subroutine, 3-559

popen subroutine, 3-107

POSIX environment

specifying, 3-2

pow function, 3-340

pow subroutine, 3-341

power function, 3-341

prefresh subroutine, 3-252

printf function, 3-506

printf subroutine, 3-288

See also printw subroutine

See also vprintf subroutine

conversion specification list, 3-506

printing date, 3-509

restrictions, 3-508

System V and, 3-289, 3-509

printw subroutine, 3-253, 3-528

priority

setting, 3-99

process

getting limits, 3-182

getting resource accounting, 3-190

getting time-accounting information, 3-178

initiating I/O, 3-107

limiting resource consumption, 3-188

setting limits, 3-182

suspending, 3-104, 3-146

terminating after flushing pending output, 3-43

terminating with core dump, 3-6

process group

setting, 3-127

program

debugging, 3-123, 3-142

profiling, 3-95

specifying addresses, 3-37, 3-38

verifying, 3-9

Program Management

pgm_exit, 3-431

protocols file

getting entry, 3-377

pseudorandom number

generating, 3-31

psignal subroutine, 3-108

ptrace system call

See also exect subroutine

putc subroutine, 3-510

putchar subroutine, 3-510

putenv subroutine, 3-109

putp subroutine, 3-254

putpwent subroutine, 3-110

puts subroutine, 3-511

putw subroutine, 3-510

Q

qsort subroutine (standard C), 3-111

queue

removing elements, 3-77

R

rand subroutine (standard C), 3-112

See also random subroutine

random number

See also pseudorandom number

random number generator, 3-112, 3-113

random subroutine, 3-113

raw subroutine, 3-255, 3-528

rcmd subroutine, 3-562

readdir subroutine, 3-27

real group ID

setting, 3-129

real user ID

setting, 3-129

realloc subroutine (special library), 3-555

realloc subroutine (standard C), 3-84, 3-86

re_comp subroutine, 3-115

re_exec subroutine, 3-115

refresh macro, 3-256

refresh subroutine, 3-528

regex keyword, 3-115

regular expression handler, 3-115

Remote Procedure Call

rpc_listen, 3-448

Remote Remote Procedure Call

rrpc_inq_stats, 3-474

Remote Remote Procedure Call Interface

- `rrpc_inq_interface`, 3-473
- `rrpc_shutdown`, 3-476
- remove**
 - files, 3-116
- remove files**, 3-116
- remque subroutine**, 3-77
- reset_prog_mode subroutine**, 3-258
- reset_shell_mode subroutine**, 3-258
- resetty subroutine**, 3-257, 3-528
- resolver routines**, 3-117
 - and BIND servers, 3-117
- restartterm subroutine**, 3-259
- rewind subroutine**, 3-503
- rewinddir subroutine**, 3-27
- rexec subroutine**, 3-564
- rindex subroutine (standard C)**, 3-147
- rint subroutine**, 3-345
- round-to-nearest function**, 3-343, 3-345
- rpc_allow_remote_shutdown**, 3-434
- rrsvport subroutine**, 3-562
- rrpc_inq interface**
 - Remote Remote Procedure Call Interface, 3-473
- ruserok subroutine**, 3-562

S

- savetty subroutine**, 3-257, 3-528
- scandir subroutine**, 3-120
- scanf function**, 3-512, 3-515
 - conversion specification list, 3-512
- scanf subroutine**, 3-290
 - conversion specification list, 3-290
 - restrictions, 3-515
- scanw subroutine**, 3-260, 3-528
- screen**
 - updating, 3-193, 3-528
- scroll subroutine**, 3-261, 3-528
- scrollok subroutine**, 3-262, 3-528
- seed48 subroutine**, 3-31
- seekdir subroutine**, 3-27
- sendind data**
 - normal, 3-613
- sending data**
 - expedited, 3-613
- services file**
 - getting entry, 3-379
- setauthfile subroutine**, 3-51
- setbuf subroutine**, 3-517
- setbuffer subroutine**, 3-517
- setegid subroutine**, 3-129
- seteuid subroutine**, 3-129
- setfsent subroutine**, 3-534
- setgid subroutine**, 3-129
- setgrent subroutine**, 3-56
- sethostent subroutine**, 3-373
- setitimer system call**
 - See also* pause subroutine
- setjmp subroutine**, 3-123
- setlinebuf subroutine**, 3-517
- setlocale subroutine**, 3-124
- setnetent subroutine**, 3-375
- setnetgrent subroutine**, 3-626
- setpgid function**, 3-127
- setprotoent subroutine**, 3-377
- setpwent subroutine**, 3-65
- setpwnfile subroutine**, 3-65
- setrgid subroutine**, 3-129
- setrpcent subroutine**, 3-67
- setruid subroutine**, 3-129
- setscrreg subroutine**, 3-263
- setservent subroutine**, 3-379
- setstate subroutine**, 3-113
- set_term subroutine**, 3-265
- setterm subroutine**, 3-528
- settyent subroutine**, 3-71
- setuid subroutine**, 3-129
- setupterm subroutine**, 3-264
- setvbuf subroutine**, 3-517
- shell command**
 - issuing, 3-176
- sigaction system call**, 3-131
 - diagnostics, 3-131
- sigaddset subroutine**, 3-143
- sigblock system call**
 - See also* sigsuspend system call

- sigdelset subroutine**, 3-143
- sigemptyset subroutine**, 3-143
- sigfillset subroutine**, 3-143
- siginterrupt**, 3-132
- sigismember subroutine**, 3-143
- siglongjmp subroutine**, 3-142
- signal**, 3-143
 - blocking, 3-140
 - changing action, 3-133, 3-137
 - releasing blocked, 3-145
- signal handler**
 - assigning, 3-131
- signal message**
 - getting, 3-108
- signal subroutine (standard C)**, 3-133, 3-137
 - mapping hardware traps to signals, 3-139
 - return value, 3-136, 3-138
 - signal list, 3-133, 3-137
- sigprocmask system call**, 3-140
- sigsetjmp subroutine**, 3-142
- sigsuspend system call**, 3-145
- sigvec system call**
 - See also* signal subroutine (standard C)
- sin subroutine**, 3-362
- sin subroutine (math)**, 3-364
 - restrictions, 3-364
 - return value, 3-364
 - System V and, 3-364
- sine function**, 3-364
- sinh subroutine (math)**, 3-367
- sleep subroutine (standard C)**, 3-146
- snmpext**
 - library routines, 3-383
- Socket Call**
 - socket to numeric name, 3-484
- sort routine**, 3-111
- space subroutine**, 3-559
- special file**
 - creating, 3-90
- sprintf function**, 3-506
- sprintf subroutine**, 3-288
- sqrt subroutine (math)**, 3-341
- square root function**, 3-341
- strand subroutine**, 3-112
- strand48 subroutine**, 3-31
- strandom subroutine**, 3-113
- sscanf function**, 3-512
- sscanf subroutine**, 3-290
- standard buffered input/output package**
 - See* stdio macro package
- standard IO**
 - subroutines for, 3-510
- standend subroutine**, 3-203, 3-528
- standout subroutine**, 3-203, 3-528
- stdio macro package**, 3-493
 - diagnostics, 3-494
- stime system call**, 3-160
 - See also* gettimeofday system call
- store subroutine**, 3-530
- storeauthent subroutine**, 3-51
- strcat subroutine**, 3-147
 - See also* bcopy subroutine
 - diagnostics, 3-147
- strchr subroutine**, 3-147
- strcmp subroutine**, 3-147
- strcoll subroutine**, 3-163
- strcpy subroutine**, 3-147
- strcspn subroutine**, 3-147
- stream**
 - appending, 3-502
 - assigning buffering to, 3-517
 - closing, 3-497
 - defined, 3-493
 - flushing, 3-497
 - getting character from, 3-504
 - getting string from, 3-505
 - getting word from, 3-504
 - opening, 3-500
 - pushing character back, 3-522
 - putting character on, 3-510
 - putting string on, 3-511
 - putting word on, 3-510
 - reading, 3-502
 - repositioning, 3-503
 - returning to remote command, 3-562, 3-564
 - status queries, 3-498

strftime subroutine, 3-164
string
 operations on null-terminated, 3-147
 operations on variable-length, 3-14
string keyword, 3-147
strlen subroutine, 3-147
strncat subroutine, 3-147
strncmp subroutine, 3-147
strncpy subroutine, 3-147
strpbrk subroutine, 3-147
strrchr subroutine, 3-147
strspn subroutine, 3-147
strtod subroutine, 3-10
strtok subroutine, 3-147
strtol subroutine, 3-10
strxfrm subroutine, 3-169
stty subroutine, 3-170
 See also ioctl system call
subroutine
 See also specific subroutines
 executing at specified time, 3-8
subroutines
 international, 3-277
subwin subroutine, 3-266, 3-528
swab subroutine, 3-171
sysconf subroutine, 3-174
sys_errlist subroutine, 3-105
syslog subroutine, 3-175
sys_nerr subroutine, 3-105
sys_siglist subroutine, 3-108
system log
 controlling, 3-175
system subroutine (standard C), 3-176
System V environment
 specifying, 3-2

T

t_accept system call, 3-569
t_alloc system call, 3-572
tan subroutine, 3-362
tan subroutine (math), 3-364
tanh subroutine (math), 3-367
t_bind system call, 3-574
t_close system call, 3-577
 See also t_open system call
t_connect system call, 3-578
tdelete subroutine, 3-179
telldir subroutine, 3-27
tempnam subroutine, 3-520
temporary file
 creating, 3-519
 naming, 3-520
termcap keyword, 3-565
terminal
 finding name, 3-181
 generating file name for, 3-495
 getting state, 3-170
 independent operation routines, 3-565
 setting state, 3-170
 updating screen, 3-193, 3-528
t_error system function, 3-581
tfind subroutine, 3-179
t_free system call, 3-583
 See also t_alloc system call
tgetent subroutine, 3-267, 3-565
 See also curses package
 See also termcap file
tgetflag subroutine, 3-267, 3-565
tgetnum subroutine, 3-267, 3-565
t_getstate system call, 3-588
tgetstr subroutine, 3-267, 3-565
tgoto subroutine, 3-267, 3-565
time
 converting to ASCII, 3-20
 getting, 3-177
 reporting, 3-15
 setting, 3-160
time subroutine (standard C), 3-177
times subroutine, 3-178
timezone subroutine, 3-20
t_listen system call, 3-590
t_look system call, 3-592
tmpfile subroutine, 3-519
 See also mktemp subroutine
 See also tmpname subroutine

- tmpnam subroutine**, 3-520
 - See also* mktemp subroutine
 - See also* tmpfile subroutine
- toacsii subroutine**, 3-16
- _tolower subroutine**, 3-16
- tolower subroutine (standard C)**, 3-16
- t_open system call**, 3-596
- t_optmgmt system call**, 3-598
- touchwin subroutine**, 3-268, 3-528
- _toupper subroutine**, 3-16
- toupper subroutine**, 3-16
- tparm subroutine**, 3-269
- tputs subroutine**, 3-270, 3-565
- traceon subroutine**, 3-271
- transport connection**
 - abortive release, 3-616
 - orderly release, 3-618
- transport endpoint**
 - closing, 3-577
 - disabling, 3-624
 - initializing, 3-596
 - polling for asynchronous events, 3-592
 - protocol address, 3-574
 - returning current event, 3-592
- transport library**
 - synchronizing, 3-622
- transport protocol**
 - returning characteristics, 3-587
- transport provider**
 - returning current state, 3-588
- transport user**
 - accepting connection, 3-569
 - protocol options, 3-598
 - requesting a connection, 3-578
- t_rcv system call**, 3-600
- t_rcvconnect system call**, 3-602
- t_rcvdis system call**, 3-605
- t_rcvrel. system call**, 3-607
- t_rcvudata system call**, 3-609
- truncation**, 3-343
- tsearch subroutine**, 3-179
 - diagnostics, 3-180
 - restrictions, 3-180
- t_snd system call**, 3-613
- t_snddis system call**, 3-616
- t_sndrel system call**, 3-618
- t_sndudata system call**, 3-620
- t_sync system call**, 3-622
- t_tcvuderr system call**, 3-611
- ttyname subroutine**, 3-181
 - See also* ctermid subroutine
- ttys file**
 - accessing data, 3-71
 - field definitions, 3-71
- ttyslot subroutine**, 3-181
- t_unbind system call**, 3-624
- twalk subroutine**, 3-179
- typeahead subroutine**, 3-272
- tzet subroutine**, 3-20

U

- ulimit system call**, 3-182
- unctrl macro**, 3-273
- unctrl subroutine**, 3-528
- ungetc subroutine**, 3-522
- utime function**, 3-183

V

- valloc subroutine**, 3-185
- varargs subroutine**, 3-186
 - See also* vprintf subroutine
 - example, 3-186
 - restrictions, 3-187
- vfprintf subroutine**, 3-292, 3-523
- vidattr subroutine**, 3-274
- vidputs subroutine**, 3-274
- vlimit subroutine**, 3-188
 - See also* getrlimit system call
 - restrictions, 3-189
- /vmunix file**
 - examining name list, 3-100, 3-101
- vprintf subroutine**, 3-292, 3-523
 - writing an error routine and, 3-523e
- vsprintf subroutine**, 3-292

vtimes subroutine, 3-190
See also getrusage system call

W

waddch subroutine, 3-200, 3-528
waddstr subroutine, 3-202, 3-528
wattroff subroutine, 3-203
wattron subroutine, 3-203
wattrset subroutine, 3-203
wclear subroutine, 3-209, 3-528
wclrtoeb subroutine, 3-211, 3-528
wclrtoeol subroutine, 3-212, 3-528
wdelch subroutine, 3-215, 3-528
wdeleteln subroutine, 3-216, 3-528
werase subroutine, 3-221, 3-528
wgetch subroutine, 3-224, 3-528
wgetstr subroutine, 3-226, 3-528
winch macro, 3-232
winch subroutine, 3-528
window
 defined, 3-193
winsch macro, 3-234
winsch subroutine, 3-528
winserln subroutine, 3-235, 3-528
wmove subroutine, 3-242, 3-528
wnoutrefresh subroutine, 3-275
working directory
 getting pathname, 3-53, 3-73
wprintw subroutine, 3-253, 3-528
wrefresh subroutine, 3-256, 3-528
 See also wnoutrefresh subroutine
wscanw subroutine, 3-260, 3-528
wsetscrreg subroutine, 3-263
wstandend subroutine, 3-203, 3-528
wstandout subroutine, 3-203, 3-528

X

X/Open Transport Interface
 introduction, 3-567

Y

y0 subroutine, 3-336
y1 subroutine, 3-336
yn subroutine, 3-336
YP client interface, 3-628
YP service
 library function, 3-625
yp_all subroutine, 3-628
yp_bind subroutine, 3-628
ypclnt keyword, 3-628
yperr_string subroutine, 3-628
yp_first subroutine, 3-628
yp_get_default_domain subroutine, 3-628
yp_master subroutine, 3-628
yp_match subroutine, 3-628
yp_next subroutine, 3-628
yp_order subroutine, 3-628
yppasswd subroutine, 3-632
ypprot_err subroutine, 3-628
yp_unbind subroutine, 3-628

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-234-1998 using a 1200- or 2400-baud modem from anywhere in the USA, Canada, or Puerto Rico. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital Subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	_____	Local Digital subsidiary or approved distributor
Internal*	_____	SSB Order Processing - WMO/E15 or Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

* For internal orders, you must submit an Internal Software Order Form (EN-01740-07).

Reader's Comments

Reference Pages Section 3: Library Routines
ULTRIX
AA-LY16B-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:

	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

What would you like to see more/less of? _____

What do you like best about this manual? _____

What do you like least about this manual? _____

Please list errors you have found in this manual:

Page	Description
------	-------------

_____	_____
-------	-------

_____	_____
-------	-------

_____	_____
-------	-------

_____	_____
-------	-------

_____	_____
-------	-------

Additional comments or suggestions to improve this manual:

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

----- Do Not Tear - Fold Here and Tape -----

digital™



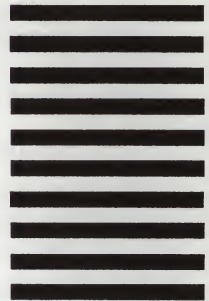
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3-2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062-9987



----- Do Not Tear - Fold Here -----

Cut
Along
Dotted
Line

Reader's Comments

Reference Pages Section 3: Library Routines
ULTRIX
AA-LY16B-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

Please rate this manual:

	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Completeness (enough information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Clarity (easy to understand)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Organization (structure of subject matter)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Figures (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Examples (useful)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Index (ability to find topic)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Page layout (easy to find information)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

What would you like to see more/less of? _____

What do you like best about this manual? _____

What do you like least about this manual? _____

Please list errors you have found in this manual:

Page	Description
_____	_____
_____	_____
_____	_____
_____	_____
_____	_____

Additional comments or suggestions to improve this manual:

What version of the software described by this manual are you using? _____

Name/Title _____ Dept. _____

Company _____ Date _____

Mailing Address _____

_____ Email _____ Phone _____

----- Do Not Tear - Fold Here and Tape -----

digital™



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST-CLASS MAIL PERMIT NO. 33 MAYNARD MA

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION
OPEN SOFTWARE PUBLICATIONS MANAGER
ZKO3-2/Z04
110 SPIT BROOK ROAD
NASHUA NH 03062-9987



----- Do Not Tear - Fold Here -----

Cut
Along
Dotted
Line

digital